

日 本 国 特 許 庁  
JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出 願 年 月 日                      2 0 0 2 年 1 1 月 2 8 日  
Date of Application:

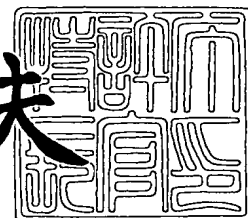
出 願 番 号                      特 願 2 0 0 2 - 3 4 5 2 3 0  
Application Number:  
[ST. 10/C]:                      [ J P 2 0 0 2 - 3 4 5 2 3 0 ]

出      願      人                      インターナショナル・ビジネス・マシーンズ・コーポレーシ  
Applicant(s):                      ョン

2 0 0 3 年    7 月 1 8 日

特許庁長官  
Commissioner,  
Japan Patent Office

今 井 康 夫



【書類名】 特許願

【整理番号】 JP9010240

【提出日】 平成14年11月28日

【あて先】 特許庁長官 殿

【国際特許分類】 H03M 13/00

【発明者】

【住所又は居所】 神奈川県大和市下鶴間 1 6 2 3 番地 1 4 日本アイ・ビー・エム株式会社 大和事業所内

【氏名】 原田 篤

【特許出願人】

【識別番号】 390009531

【氏名又は名称】 インターナショナル・ビジネス・マシーンズ・コーポレーション

【代理人】

【識別番号】 100086243

【弁理士】

【氏名又は名称】 坂口 博

【代理人】

【識別番号】 100091568

【弁理士】

【氏名又は名称】 市位 嘉宏

【代理人】

【識別番号】 100108501

【弁理士】

【氏名又は名称】 上野 剛史

【復代理人】

【識別番号】 100104880

【弁理士】

【氏名又は名称】 古部 次郎

## 【手数料の表示】

【予納台帳番号】 081504

【納付金額】 21,000円

## 【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 9706050

【包括委任状番号】 9704733

【包括委任状番号】 0207860

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 データ処理方法、これを用いた通信システム、そのデータ送信装置、データ受信装置及びプログラム

【特許請求の範囲】

【請求項 1】 ネットワーク上の端末装置間でデータの送受信を行うためのデータ処理方法であって、

所定の記憶装置から送信しようとする元データのデータファイルを読み出し、当該データファイルを  $k$  個のデータからなるクラスタに分割するステップと、

前記クラスタの  $k$  個の元データに  $s$  個の畳み込みデータを加え、エンコードして  $t + s$  個のパリティ・データを生成するステップと、

前記パリティ・データのうちの  $t$  個のデータを前記元データに付加して  $k + t$  個のデータからなる送信データを作成するステップと、

所定のネットワークインターフェイスを介して、前記送信データを前記ネットワーク上の他の端末へ送信するステップとを含むことを特徴とするデータ処理方法。

【請求項 2】 前記畳み込みデータを、他のクラスタにおけるデータに依存して決定することを特徴とする請求項 1 に記載のデータ処理方法。

【請求項 3】 前記送信データを作成するステップでは、所定の前記クラスタに関して、生成された  $t + s$  個のパリティ・データのうち  $t' > t$  なる  $t'$  個のデータを前記元データに付加して送信データを作成することを特徴とする請求項 1 に記載のデータ処理方法。

【請求項 4】 ネットワーク上の端末装置間でデータの送受信を行うためのデータ処理方法であって、

所定の記憶装置から送信しようとする元データのデータファイルを読み出し、当該データファイルをクラスタに分割するステップと、

前記クラスタの元データをエンコードし、他のクラスタのデータに関する情報を含むパリティ・データを生成するステップと、

前記パリティ・データを前記元データに付加して送信データを作成するステップと、

所定のネットワークインターフェイスを介して、前記送信データを前記ネットワーク上の他の端末へ送信するステップとを含むことを特徴とするデータ処理方法。

【請求項 5】 前記パリティ・データを生成するステップでは、前記元データに他のクラスタにおけるパリティ・データの一部を付加してエンコードすることにより、当該他のクラスタのパリティ・データに関する情報を含むパリティ・データを生成することを特徴とする請求項 4 に記載のデータ処理方法。

【請求項 6】 前記パリティ・データの少なくとも一部は、他のクラスタにおける元データをエンコードする際に当該元データに付加するデータと同一であることを特徴とする請求項 4 に記載のデータ処理方法。

【請求項 7】 ネットワーク上の端末装置間でデータの送受信を行うためのデータ処理方法であって、

クラスタに分割され、各クラスタの元データとパリティ・データとを含むデータ列を受信するステップと、

通信中に所定のクラスタに関してデータの喪失があった場合に、当該所定のクラスタにおける喪失しなかったデータをデコードして、元データと、前記パリティ・データを生成するために当該元データに追加された畳み込みデータとを復元するステップと、

通信中のデータの喪失により前記所定のクラスタのデータ数が前記元データ及び前記畳み込みデータを復元するために必要な必要数に満たない場合に、他のクラスタの復元データに基づいて当該所定のクラスタのデータを補完してデコードし、前記元データ及び前記畳み込みデータを復元するステップと、

受信または復元された各クラスタの元データを連結してデータファイルを作成し、所定の記憶装置に格納するステップとを含むことを特徴とするデータ処理方法。

【請求項 8】 前記所定のクラスタのデータ数が前記必要数に満たない場合、前記元データをエンコードした際のエンコード順において当該所定のクラスタの直前のクラスタにおける復元された前記元データ及び前記畳み込みデータをエンコードして得られたデータ、または当該エンコード順において当該所定のクラ

スタの直後のクラスタにおける復元された前記畳み込みデータ、またはその両方を用いて当該クラスタのデータを補完することを特徴とする請求項 7 に記載のデータ処理方法。

【請求項 9】 ネットワークを介して端末装置間でデータ交換を行う通信システムにおいて、

送信側の端末装置は、送信しようとする元データのデータファイルをクラスタに分割し、各クラスタの元データに他のクラスタにおける所定のデータを付加してエンコードすることによりパリティ・データを生成し、当該元データに当該パリティ・データを付加して生成された送信データを受信側の端末装置に送信し、

受信側の端末装置は、前記送信データを受信し、当該送信データの一部が通信中に喪失した場合、前記クラスタごとに前記元データを復元することを特徴とする通信システム。

【請求項 10】 前記受信側の端末装置は、受信した前記送信データのうち所定のクラスタにおけるデータ数が通信中のデータの喪失により前記元データを復元するために必要な必要数に満たない場合に、他のクラスタの復元データに基づいて当該所定のクラスタのデータを補完して前記元データを復元することを特徴とする請求項 9 に記載の通信システム。

【請求項 11】 ネットワークを介してデータ送信を行うデータ送信装置において、

元データのデータファイルを格納したファイル格納部と、

前記ファイル格納部から送信する元データのデータファイルを読み出してクラスタに分割し、当該クラスタごとに元データに他のクラスタにおける所定のデータを付加してエンコードすることによりパリティ・データを生成し、当該パリティ・データと当該元データとを含む送信データを作成する送信データ作成部と、

前記送信データ作成部にて作成された前記送信データを送信する送信制御部とを備えることを特徴とするデータ送信装置。

【請求項 12】 前記送信データ作成部は、所定のクラスタに関して、前記元データをエンコードした際のエンコード順において当該所定のクラスタの直前のクラスタにおけるパリティ・データを前記元データに付加してエンコードする

ことを特徴とする請求項 11 に記載のデータ送信装置。

【請求項 13】 ネットワークを介して送信されたデータを受信するデータ受信装置において、

クラスタに分割され、各クラスタの元データとパリティ・データとを含むデータ列を受信する受信制御部と、

通信中に所定のクラスタに関してデータの喪失があった場合に、当該所定のクラスタにおける喪失しなかったデータをデコードして、元データと、前記パリティ・データを生成するために当該元データに追加された畳み込みデータとを復元し、当該データの喪失により前記所定のクラスタのデータ数が前記元データ及び前記畳み込みデータを復元するために必要な必要数に満たない場合に、他のクラスタの復元データに基づいて当該所定のクラスタのデータを補完してデコードし、前記元データ及び前記畳み込みデータを復元するデータ復元部と、

受信または復元された前記元データを連結して得られたデータファイルを格納するファイル格納部と

を備えることを特徴とするデータ受信装置。

【請求項 14】 前記データ復元部は、前記所定のクラスタのデータ数が前記必要数に満たない場合、前記元データをエンコードした際のエンコード順において当該所定のクラスタの直前のクラスタにおける復元された前記元データ及び前記畳み込みデータをエンコードして得られたデータ、または当該エンコード順において当該所定のクラスタの直後のクラスタにおける復元された前記畳み込みデータ、またはその両方を用いて当該クラスタのデータを補完することを特徴とする請求項 13 に記載のデータ受信装置。

【請求項 15】 ネットワークに接続されたコンピュータを制御して、データの送信処理を行うプログラムであって、

所定の記憶装置から送信しようとする元データのデータファイルを読み出し、当該データファイルを  $k$  個のデータからなるクラスタに分割する第 1 の処理と、

前記クラスタの  $k$  個の元データに  $s$  個の畳み込みデータを加え、エンコードして  $t + s$  個のパリティ・データを生成する第 2 の処理と、

前記パリティ・データのうちの  $t$  個のデータを前記元データに付加して  $k + t$

個のデータからなる送信データを作成する第3の処理と、

所定のネットワークインターフェイスを介して、前記送信データを送信する第4の処理と

を前記コンピュータに実行させることを特徴とするプログラム。

【請求項16】 前記プログラムによる前記第2の処理では、所定のクラスタにおける前記畳み込みデータとして、直前のクラスタに関して生成された前記パリティ・データにおける所定の  $s$  個のデータを用いることを特徴とする請求項15に記載のプログラム。

【請求項17】 所定のクラスタに関して、前記第2の処理により生成された  $t + s$  個のパリティ・データのうち、前記第3の処理で抽出した  $t$  個のデータ以外のデータを送信する処理を前記コンピュータにさらに実行させることを特徴とする請求項15に記載のプログラム。

【請求項18】 ネットワークに接続されたコンピュータを制御して、当該ネットワークを介して送信されたデータの受信処理を行うプログラムであって、  
クラスタに分割され、各クラスタの元データとパリティ・データとを含むデータ列を受信する第1の処理と、

通信中に所定のクラスタに関してデータの喪失があった場合に、当該所定のクラスタにおける喪失しなかったデータをデコードして、元データと、前記パリティ・データを生成するために当該元データに追加された畳み込みデータとを復元する第2の処理と、

通信中のデータの喪失により前記所定のクラスタのデータ数が前記元データ及び前記畳み込みデータを復元するために必要な必要数に満たない場合に、他のクラスタの復元データに基づいて当該所定のクラスタのデータを補完してデコードし、前記元データ及び前記畳み込みデータを復元する第3の処理と、

受信または復元された各クラスタの元データを連結してデータファイルを作成し、所定の記憶装置に格納する第4の処理と  
を前記コンピュータに実行させることを特徴とするプログラム。

【請求項19】 前記プログラムによる前記第3の処理では、前記元データをエンコードした際のエンコード順において当該所定のクラスタの直前のクラス



タにおける復元された前記元データ及び前記畳み込みデータをエンコードして得られたデータ、または当該エンコード順において当該所定のクラスタの直後のクラスタにおける復元された前記畳み込みデータ、またはその両方を用いて当該クラスタのデータを補完することを特徴とする請求項 18 に記載のプログラム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、ネットワークにおけるデータ通信に関し、特に F E C によりデータ通信における信頼性を向上させる技術に関する。

【0002】

【従来の技術】

インターネットに代表されるコンピュータネットワークの普及により、ネットワークを介して行われるデータ通信の信頼性を向上させることが重要な課題となっている。

従来、データ通信の信頼性を高める手段としては、データ通信が失敗した場合（データが壊れたり喪失したりした場合、以下、喪失と総称する）に受信側端末から送信側端末へ自動的にデータの再送を要求する A R Q (Automatic Repeat r eQuest: 再送要求) や、データ喪失の可能性を予め想定して送信データを冗長に構成する F E C (Forward Error Correction)、これらの組合せなどがある。A R Q が、受信側端末から送信側端末へ要求を行うために双方向通信であることを前提としているのに対し、F E C は冗長な送信データから元データを復元するので 1 方向通信にて利用することが可能である。

【0003】

F E C により信頼性の向上を図る手法として、受信者において、正しく受信されたデータから消失データを復元することが可能なイレーシャ・コード (Erasure Code) を用いるものがある（例えば、非特許文献 1 参照。また、イレーシャ・コードの理論的な根拠については、例えば、非特許文献 2 参照。）。この手法では、送信者は、送信しようとする元データにパリティ・データを付加して送信データを作成し、送信する。受信者は、受信した送信データから元データを抽出す

るが、この際、元データとパリティ・データの合計のデータ量が一定以上あれば、送信データの一部が喪失されていても元データを復元することができる。

#### 【0004】

このイレシャ・コードを用いたFECによるデータ通信方法として、例えば、特許文献1に開示された従来技術がある。

図11は、特許文献1に開示された従来技術によるデータ通信方法を説明する図である。

同文献に開示された技術によれば、図11に示すように、次の手順でデータ通信を行う。

(1) データファイル(元データ)を $k$ 個のデータ(図示の例では6個)からなるクラスタに分割する(図示の例では、データファイルは8個のクラスタに分割されている。なお、クラスタの文言は、送信しようとする元データの分割単位を表す。したがって、以下の説明でクラスタと言った場合、その内容は、対応する元データである場合、元データとパリティ・データとの組合せである場合などがある。また、データ数の単位は、例えば1Kbyte、8Kbyte等とすることができる。

(2) 送信者1101は、個々のクラスタに対して、 $k$ 個の元データから $t$ 個のパリティ・データ(図示の例では4個)を生成(エンコード)し、元データと共に(すなわち $k+t$ 個のデータからなるデータ列を)送信する。

(3) 受信者1102は、かかるデータ列を受信するが、通信中にデータ列の一部が喪失していた場合、このデータ列中の任意の $k$ 個から該当クラスタの元データを復元(デコード)する。

図11に示す例では、元データとパリティ・データを合わせて6個のデータがあれば、言い換えれば4個までは喪失しても、元データを復元できる。

#### 【0005】

図12は、特許文献1に開示された従来技術によるパリティ・データの生成方法を示す図である。

同文献に開示された技術では、図12に示すように、各クラスタの元データからそれぞれ独立にパリティ・データが生成される。したがって、個々のクラスタ

において元データを復元する場合、当該クラスタにおけるパリティ・データを含むデータ列からのみ復元される。

すなわち、この技術では、データの受信成功には各クラスタについて  $k$  個以上のデータを受信することが必要となる。また、データの配信成功には全ての受信者による受信成功が必要である。

#### 【0006】

##### 【特許文献1】

米国特許明細書第 6012159 号明細書

##### 【非特許文献1】

L. Rizzo, “Effective erasure codes for reliable computer communication protocols”、[online]、April 1997、ACM Computer Communication Review、[平成14年8月30日検索]、インターネット<URL: <http://citeseer.nj.nec.com/rizzo97effective.html>>

##### 【非特許文献2】

Johannes Blomer、Malik Kalfane、Richard Karp、Marek Karpinski、Michael Luby、David Zuckerman, “An XOR-Based Erasure-Resilient Coding Scheme”、[online]、1995、[平成14年8月30日検索]、インターネット<URL: <http://citeseer.nj.nec.com/84162.html>>

#### 【0007】

##### 【発明が解決しようとする課題】

しかし、上記従来のイレシャ・コードを用いたデータ通信方式は、クラスタ内の元データの数が  $k$  個である場合、 $k + t$  個（元データ + パリティ・データ）のデータのうち  $t$  個以上のデータが喪失した場合（すなわち受信できたデータが  $k$  個に満たない場合）には元データを復元することができなかった。

また、上述したように、データ通信では、データの配信成功には全ての受信者による受信成功が必要であることから、配信成功確率は受信者数およびクラスタ数が増加するのに伴って減少する。

#### 【0008】

ここで、データの送信が失敗した場合に、受信者から送信者へマニュアル操作

にて当該データの再送要求がなされ、再送信する場合の処理コストを考える。この場合、各受信者へ個別にデータ送信が行われるユニキャスト配信であれば、再送信の処理コストは再送信を行う受信者数に比例するため、一般に初期配信の処理コストよりも小さい。一方、同報配信（ブロードキャストやマルチキャスト）の場合、再送信の処理コストは受信者数によらないため、初期配信の処理コストと同等である。このため、同報配信において特に高い配信成功確率が要求されることとなる。

#### 【0009】

特許文献1に開示された従来のデータ通信方法では、配信成功確率を高めるためには、クラスタにおけるパリティ・データのデータ数 $t$ または元データのデータ数 $k$ の片方または双方の値を大きくする。しかしながら、パリティ・データの割合（冗長度）を増やすと、送信するデータ量及び送信時間が増加してしまう。また、元データのデータ数 $k$ を大きくすると、エンコード及びデコードに要する計算量の大きな増加を招く。

したがって、FECによるデータ通信では、これらの点に改善の余地を見出すことができる。

#### 【0010】

そこで本発明は、FECによるデータ通信において、配信成功確率を高め、データ通信の信頼性の向上を図ることを目的とする。

また、本発明は、FECによるデータ通信における冗長度（パリティ・データの割合）を低く抑えることを他の目的とする。

#### 【0011】

##### 【課題を解決するための手段】

上記の目的を達成する本発明は、次のようなデータ処理方法として実現することができる。すなわち、このデータ処理方法は、所定の記憶装置から送信しようとする元データのデータファイルを読み出し、このデータファイルを同じ大きさの $k$ 個のデータからなるクラスタに分割するステップと、このクラスタの $k$ 個の元データに $s$ 個の畳み込みデータを加え、エンコードして $t + s$ 個のパリティ・データを生成するステップと、このパリティ・データのうちの $t$ 個のデータを元

データに付加して  $k + t$  個のデータからなる送信データを作成するステップと、所定のネットワークインターフェイスを介して、この送信データをネットワーク上の他の端末へ送信するステップとを含むことを特徴とする。

#### 【0012】

ここでこの畳み込みデータを、他のクラスタにおけるデータに依存して決定する。より詳細には、所定のクラスタにおける畳み込みデータとして、パリティ・データ生成の際のエンコード順における直前のクラスタに関して生成されたパリティ・データにおける所定の  $s$  個のデータを用いる。また、パリティ・データ生成の際のエンコード順における最後のクラスタにおける受信成功確率の低下を防ぐため、かかる最後のクラスタに関しては、特に生成した  $t + s$  個のパリティ・データの全てを送信するようにしても良い。あるいは  $t + s$  個のパリティ・データのうち、 $t' > t$  なる  $t'$  個のデータを前記元データに付加して送信データを作成するようにしても良い。

#### 【0013】

また本発明は、次のようなデータ処理方法としても実現される。すなわち、このデータ処理方法は、所定の記憶装置から送信しようとする元データのデータファイルを読み出し、このデータファイルをクラスタに分割するステップと、このクラスタの元データと他のクラスタのデータの一部で構成される畳み込みデータとをエンコードすることにより、他のクラスタのデータに関する情報を含むパリティ・データを生成するステップと、このパリティ・データを元データに付加して送信データを作成するステップと、所定のネットワークインターフェイスを介して、この送信データをネットワーク上の他の端末へ送信するステップとを含むことを特徴とする。

#### 【0014】

ここで、パリティ・データを生成する際には、元データに他のクラスタにおけるパリティ・データの一部を付加してエンコードする。これにより、生成されたパリティ・データには、この他のクラスタのパリティ・データに関する情報が含まれることとなる。また、このパリティ・データの一部もさらに他のクラスタの元データをエンコードする際に付加されることとなるので、生成されたパリティ

・データには、かかるさらに他のクラスタのデータに関する情報も含まれることとなる。

#### 【0015】

さらに本発明は、以上のようにして生成された送信データを受信する場合の、次のようなデータ処理方法としても実現される。すなわち、このデータ処理方法は、クラスタに分割され、各クラスタの元データとパリティ・データとを含むデータ列を受信するステップと、通信中に所定のクラスタに関してデータの喪失があった場合に、所定のクラスタにおける喪失しなかったデータをデコードして、元データと、パリティ・データを生成するためにこの元データに追加された畳み込みデータとを復元するステップと、通信中のデータの喪失によりこの所定のクラスタのデータ数が元データ及び畳み込みデータを復元するために必要な必要数に満たない場合に、他のクラスタの復元データに基づき、この所定のクラスタのデータを補完してデコードし、元データ及び畳み込みデータを復元するステップと、受信または復元された各クラスタの元データを連結してデータファイルを作成し、所定の記憶装置に格納するステップとを含むことを特徴とする。

このクラスタのデータの補完には、元データをエンコードした際のエンコード順において処理対象であるクラスタの直前のクラスタにおける復元された元データ及び畳み込みデータをエンコードして得られたデータ、またはこのエンコード順において直後のクラスタにおける復元された前記畳み込みデータ、あるいはその両方を用いることができる。

#### 【0016】

また、上記の目的を達成するさらに他の本発明は、次のように構成されたことを特徴とする通信システムとしても実現することができる。この通信システムにおいて、送信側の端末装置は、送信しようとする元データのデータファイルをクラスタに分割し、各クラスタの元データに他のクラスタにおける所定のデータを付加してエンコードすることによりパリティ・データを生成し、このパリティ・データを元データに付加して生成された送信データを受信側の端末装置に送信する。また、受信側の端末装置は、この送信データを受信し、この送信データの一部が通信中に喪失した場合、クラスタごとに元データを復元する。ここで、受信

した所定のクラスタにおける送信データのデータ数が通信中のデータの喪失により元データを復元するために必要な必要数に満たない場合に、他のクラスタの復元データに基づいてこのクラスタのデータを補完して元データを復元する。

#### 【0017】

さらに詳細には、この送信側の端末装置であるデータ送信装置は、元データのデータファイルを格納したファイル格納部と、このファイル格納部から送信する元データのデータファイルを読み出してクラスタに分割し、このクラスタごとに元データに他のクラスタ（エンコード順における直前のクラスタ）における所定のデータを付加してエンコードすることによりパリティ・データを生成し、このパリティ・データと元データとを含む送信データを作成する送信データ作成部と、この送信データ作成部にて作成された送信データを送信する送信制御部とを備えることを特徴とする。

#### 【0018】

また、この受信側の端末装置であるデータ受信装置は、クラスタに分割され、各クラスタの元データとパリティ・データとを含むデータ列を受信する受信制御部と、通信中に所定のクラスタに関してデータの喪失があった場合に、この所定のクラスタにおける喪失しなかったデータをデコードして、元データと、パリティ・データを生成するためにこの元データに追加された畳み込みデータとを復元し、通信中のデータの喪失によりこの所定のクラスタのデータ数が元データ及び畳み込みデータを復元するために必要な必要数に満たない場合に、他のクラスタの復元データに基づいてこの所定のクラスタのデータを補完してデコードし、元データ及び畳み込みデータを復元するデータ復元部と、受信または復元された各クラスタの元データを連結して得られたデータファイルを格納するファイル格納部とを備えることを特徴とする。

#### 【0019】

さらに本発明は、コンピュータを制御して上述したデータ処理方法の各ステップに対応する処理を実行させるプログラムや、上記のデータ送信装置またはデータ受信装置としてコンピュータを機能させるプログラムとして実現される。このプログラムは、磁気ディスクや光ディスク、半導体メモリ、その他の記録媒体に

格納して配布したり、ネットワークを介して配信したりすることにより提供することができる。

### 【0020】

#### 【発明の実施の形態】

以下、添付図面に示す実施の形態に基づいて、この発明を詳細に説明する。

図1は、本実施の形態によるデータ通信が行われるネットワークシステムの概略構成を示す図である。

図1に示すネットワークシステムは、インターネットなどのネットワーク30に、送信側端末10と受信側端末20とが接続されて構成されている。送信側端末10及び受信側端末20は、ネットワーク30を介してパケット通信によるデータ交換を行う機能を備えたコンピュータ、PDA(Personal Digital Assistant)、携帯電話、その他の情報通信機器(端末装置)にて実現される。なお、送信側端末10及び受信側端末20は、所定のデータを送信しようとするユーザ(送信者)が使用する端末と、当該データを受信する端末とを意味する。したがって、ネットワーク30に接続された各端末装置が、通信ごとに送信側端末10にも受信側端末20にもなり得る。また図示のように、受信側端末20は複数存在し、個別に送信する他、同報配信にて一斉に送信することが可能である。

### 【0021】

図2は、図1に示した送信側端末10及び受信側端末20を実現するのに好適なコンピュータ装置のハードウェア構成の例を模式的に示した図である。

図2に示すコンピュータ装置は、演算手段であるCPU(Central Processing Unit:中央処理装置)101と、M/B(マザーボード)チップセット102及びCPUバスを介してCPU101に接続されたメインメモリ103と、同じくM/Bチップセット102及びAGP(Accelerated Graphics Port)を介してCPU101に接続されたビデオカード104と、PCI(Peripheral Component Interconnect)バスを介してM/Bチップセット102に接続されたハードディスク105、ネットワークインターフェイス106及びUSBポート107と、さらにこのPCIバスからブリッジ回路108及びISA(Industry Standard Architecture)バスなどの低速なバスを介してM/Bチップセット102



に接続されたフロッピーディスクドライブ109及びキーボード／マウス110とを備える。そして、ネットワークインターフェイス106を介してネットワーク30に接続され、送信側端末10または受信側端末20として他のコンピュータ装置とでデータ交換を行う。

なお、図2は本実施の形態を実現するコンピュータ装置のハードウェア構成を例示するに過ぎず、本実施の形態を適用可能であれば、他の種々の構成を取ることができる。例えば、ビデオカード104を設ける代わりに、ビデオメモリのみを搭載し、CPU101にてイメージデータ进行处理する構成としても良いし、ATA (AT Attachment) などのインターフェイスを介してCD-ROM (Compact Disc Read Only Memory) やDVD-ROM (Digital Versatile Disc Read Only Memory) のドライブを設けても良い。

#### 【0022】

図3は、送信側端末10の機能構成を示す図である。

図3を参照すると、送信側端末10は、送信しようとする元データのデータファイルを格納したファイル格納部11と、ファイル格納部11に格納されているデータファイルから送信データを作成する送信データ作成部12と、送信データ作成部12にて作成された送信データを受信側端末20へ送信する送信制御部13とを備える。かかる構成において、ファイル格納部11は、例えば図2に示したコンピュータ装置におけるメインメモリ103やハードディスク105にて実現される。また、送信データ作成部12及び送信制御部13は、プログラム制御されたCPU101にて実現される。

CPU101を制御してこれらの機能を実現するプログラムは、磁気ディスクや光ディスク、半導体メモリ、その他の記録媒体に格納して配布したり、ネットワーク30を介して配信したりすることにより提供される。図2に示したコンピュータ装置では、このプログラムがハードディスク105に保存（インストール）された後、メインメモリ103に読み込まれ展開されて、CPU101を制御し、送信データ作成部12及び送信制御部13として機能させる。

#### 【0023】

次に、送信データ作成部12について、さらに詳細に説明する。

送信データ作成部 12 は、第 1 に、ファイル格納部 11 から送信しようとする元データのデータファイルを読み出し、k 個のデータから構成されるクラスタに分割する。データの大きさは全て同じである。クラスタを構成する上でデータに不足が生じた場合は、全て 0 で埋める等、送信側端末 10 及び受信側端末 20 の間で予め定められた方式で補う。なお、クラスタやクラスタ内の k 個のデータはファイル内での位置の順である必要はない。第 2 に、各クラスタの元データをエンコードしてクラスタごとのパリティ・データを生成し、この元データとパリティ・データとに基づいて送信データを作成する。

ここで、本実施の形態の送信データ作成部 12 は、元データのエンコードにおいて畳み込みデータを用いたエンコードを行うと共に、所定のクラスタにおける元データのエンコードを他のクラスタに依存して行う。

#### 【0024】

図 4 は、本実施の形態におけるパリティ・データの生成方法を説明する図である。

図 4 に示すように、本実施の形態では、まず元データ 411 に対して s 個の畳み込みデータ 412 を追加する。元データ 411 のデータ数を k とすれば、k + s 個のデータからなるデータ列 410 が生成される。

次に、元データ 411 に畳み込みデータ 412 を加えた k + s 個のデータをエンコードしてパリティ・データ 420 を生成する。生成されたパリティ・データ 420 の内、s 個のデータ 422 を除いた残りのデータを、元データに付加するパリティ・データ、すなわち元データと共に送信するパリティ・データ（以下、送信用パリティ・データ 421 と称す）とする。

ここで、送信用パリティ・データ 421 のデータ数を t とすれば、

$$s \leq t$$

という関係を満たす。

#### 【0025】

また、図 4 に示すように、生成されたパリティ・データ 420 の内、送信用パリティ・データ 421 以外の s 個のデータ 422 は、各クラスタをエンコードする際のエンコード順にしたがって、次のクラスタをエンコードするための畳み込

みデータ 412 となる。すなわち、所定のクラスタに着目すれば、1つ前のクラスタのパリティ・データ 420 から  $s$  個のデータ 422 を取得して、畳み込みデータ 412 として元データ 411 に加える。これにより、クラスタのエンコードは、1つ前のクラスタに依存して行われることとなる。なお、エンコード順における最初のクラスタには、1つ前のクラスタが存在しないので、予め定められた任意の  $s$  個のデータが初期値として与えられる。

#### 【0026】

以上のようにして送信データ作成部 12 により作成される送信データは、各クラスタに対して元データ 411 及び送信用パリティ・データ 421 の  $k+t$  個のデータとなる。

送信制御部 13 は、例えば図 2 に示したコンピュータ装置におけるネットワークインターフェイス 106 を制御して、上記のようにして作成された送信データを送信する。送信データの送信は、宛先の受信側端末 20 を指定して行っても良いし（ユニキャストの場合）、宛先を指定しないで行っても良い（ブロードキャストの場合）。また、送信はクラスタ毎に行う必要はなく、各クラスタから一個ずつデータを送る作業を（クラスタのデータの数だけ）繰り返したり（Interleaving）、全てのクラスタの元データを送信した後でパリティ・データを送信したりすることができる。

#### 【0027】

図 5 は、送信側端末 10 によるデータ送信時の動作を説明するフローチャートである。なお、図示の手順は、送信側端末 10 の動作の一例に過ぎず、必ずしもこの手順には限定されない。

図 5 に示すように、送信側端末 10 では、まず、送信データ作成部 12 により、ファイル格納部 11 から読み出された元データのデータファイルが、各々  $k$  個のデータからなるクラスタに分割される（ステップ 501）。ここで、パリティ・データを生成するためのエンコードを行う順にしたがって、0 から順に番号を振るものとする。

次に、送信データ作成部 12 により、クラスタ 0（1 番目のクラスタ）の元データ 411 に、予め設定された所定の初期値からなる  $s$  個の畳み込みデータ 41

2が追加される(ステップ502)。そして、この $k+s$ 個のデータ列410がエンコードされて $t+s$ 個のパリティ・データ420が生成される(ステップ503)。そして、パリティ・データ420の内の $t$ 個の送信用パリティ・データ421と $k$ 個の元データ411とで送信データが作成される(ステップ504)。

この時点で、送信制御部13により、 $k$ 個の元データ411と $t$ 個の送信用パリティ・データ421とからなる送信データが送信される(ステップ505)。送信データがエンコード順における最後のクラスタであるならば、データ送信の処理を終了する(ステップ506)。

一方、未処理のクラスタがあるならば、次に、送信データ作成部12により、パリティ・データ420中の $s$ 個のデータ422が、次のクラスタにおける畳み込みデータ412として、当該クラスタの元データ411に追加され(ステップ506、507)、ステップ503に戻って順次処理が繰り返される。

#### 【0028】

図6は、受信側端末20の機能構成を示す図である。

図6を参照すると、受信側端末20は、送信側端末10から送信された送信データを受信する受信制御部21と、受信制御部21にて受信された受信データから元データを復元するデータ復元部22と、受信制御部21にて受信されたまたはデータ復元部22にて復元された元データのデータファイルを保存するファイル格納部23とを備える。かかる構成において、ファイル格納部23は、例えば図2に示したコンピュータ装置におけるメインメモリ103やハードディスク105にて実現される。また、受信制御部21及びデータ復元部22は、プログラム制御されたCPU101にて実現される。

CPU101を制御してこれらの機能を実現するプログラムは、磁気ディスクや光ディスク、半導体メモリ、その他の記録媒体に格納して配布したり、ネットワーク30を介して配信したりすることにより提供される。図2に示したコンピュータ装置では、このプログラムがハードディスク105に保存(インストール)された後、メインメモリ103に読み込まれ展開されて、CPU101を制御し、受信制御部21及びデータ復元部22として機能させる。

## 【0029】

次に、データ復元部 22 について、さらに詳細に説明する。

本来、送信側端末 10 から送信されたデータが通信中に喪失することなく受信側端末 20 に受信されれば、受信データ中に元データが完全な形で含まれる。しかし、送信データの一部分が喪失した場合であっても、一定の範囲内であれば、データ復元部 22 により元データを復元することが可能である。

データ復元部 22 は、受信制御部 21 にて受信された受信データのうち、所定のクラスタにおいて元データの一部分が喪失している場合、当該クラスタにおける喪失していない元データとパリティ・データとをデコードし、元データを得ることができる。このデコードはクラスタごとに行われる。なお、パリティ・データのみが失われた場合、元データは無事なのでデコードの必要はない。

ここで、本実施の形態のデータ復元部 22 は、所定のクラスタの受信データをデコードするために、必用に応じて他のクラスタにおける畳み込みデータを利用する。したがって、本実施の形態による他のクラスタのデータを利用した復元を行うためには、復元可能なクラスタについては、元データの喪失の有無に関わらず復元することが必要となる場合がある。

## 【0030】

図 7 は、本実施の形態におけるデコード方法を説明する図である。

本実施の形態では、上述したように元データに畳み込みデータを加えた  $k + s$  個のデータをエンコードして  $t + s$  個のデータからなるパリティ・データ（送信用パリティ・データ +  $s$  個のデータ）を生成している。したがって、これら  $k + s + t + s$  個のデータのうち、任意の  $k + s$  個のデータをデコードすれば、元データ及び畳み込みデータを復元することができる。そこでデータ復元部 22 は、まずデコードしようとするクラスタから、元データのデータ数  $k$  に畳み込みデータのデータ数  $s$  を加算した  $k + s$  個のデータを抽出する。送信側端末 10 から送信される各クラスタのデータ数は、上記のように  $k + t$  個のデータからなるデータ列であり、 $s \leq t$  であるので、通信の際に  $t - s$  ( $= (k + t) - (k + s)$ ) 個以上のデータを喪失していなければ、 $k + s$  個のデータを抽出できる。

そして、この  $k + s$  個のデータをデコードし、図 7 に示すように、 $k$  個の元デ

ータ 711 と  $s$  個の畳み込みデータ 712 とを復元する。

#### 【0031】

ここで、所定のクラスタの受信データに関して、 $t-s$  個よりも多くのデータが喪失し、データ数が  $k+s$  個に満たない場合を考える。この場合、当該クラスタの直前または直後のクラスタの元データ及び畳み込みデータが復元可能であるならば、これらのクラスタからデータを付与することで、当該所定のクラスタ（以下、着目クラスタ）についても元データ及び畳み込みデータを復元することができる。すなわち、所定のクラスタにおける畳み込みデータと、直前のクラスタにおけるパリティ・データのうちの送信用パリティ・データ以外の  $s$  個のデータとが同一であることを利用する（図 4 参照）。

#### 【0032】

まず、図 7 (A) を参照して、着目クラスタの直前のクラスタに関して元データ及び畳み込みデータが復元できた場合について説明する。

この場合、直前のクラスタの復元データ（データ数は  $k+s$  個）を送信側端末 10 の送信データ作成部 12 と同様の手法でエンコードすれば、送信データ作成部 12 が生成したパリティ・データ 420 と同一のパリティ・データ 720 を生成することができる。すなわち、このパリティ・データ 720 のうち、送信用パリティ・データ 421 に相当する部分以外の  $s$  個のデータ 722 は、着目クラスタにおける畳み込みデータ 712 と同一である。

そこで、当該直前のクラスタの復元データをエンコードしてパリティ・データ 720 を生成し、 $s$  個のデータ 722 の部分から必要個数のデータ（補充データ）を着目クラスタに追加して、着目クラスタにおける  $k+s$  個のデータを補完する。これにより、当該着目クラスタについても元データ 711 及び畳み込みデータ 712 が復元される。

#### 【0033】

次に、図 7 (B) を参照して、着目クラスタの直後のクラスタに関して元データ及び畳み込みデータが復元できた場合について説明する。

この場合、直後のクラスタの畳み込みデータ 712 と着目クラスタのデータ 722 とは同一である。そこで、当該直後のクラスタにおける復元された畳み込み

データ 712 から必要個数のデータ（補充データ）を着目クラスタに追加して、着目クラスタにおける  $k + s$  個のデータを補完する。これにより、当該着目クラスタについても元データ 711 及び畳み込みデータ 712 が復元される。

#### 【0034】

図 8 は、以上の手法によるデコード済みのクラスタを利用したデータ補完の様子を説明する図である。

図 8 において、クラスタにおける元データ及び畳み込みデータを復元するのに必要な受信データの数（ $k + s$  個）を必要数とする（受信データの最大数は  $k + t$  個である）。図 8 に示す例では、クラスタ 0 とクラスタ 6 とは、必要数以上のデータを受信しており、単独で元データ及び畳み込みデータを復元できる。

クラスタ 1、2 は、受信データが元データの個数分であり、必要数に満たない。しかし、クラスタ 0 の復元データから順次補充データを取得して補完することにより、元データ及び畳み込みデータを復元することができる。同様に、クラスタ 4、5、7、8 も受信データが元データの個数分であり必要数に満たないが、クラスタ 6 の復元データから順次補充データを取得して補完することにより、元データ及び畳み込みデータを復元することができる。

次にクラスタ 3 は、受信データが元データの個数よりもさらに  $s$  個、すなわち必要数よりも  $2s$  個少ない。この場合でも、上記のようにしてクラスタ 2 及びクラスタ 4 の元データ及び畳み込みデータが復元されるので、これらの復元データから  $s$  個ずつ補充データを取得して補完することにより、クラスタ 3 の元データ及び畳み込みデータを復元することができる。

#### 【0035】

このように、本実施の形態では、1 つのクラスタ単独で元データを復元するためには元データの個数と畳み込みデータの個数とを加算した数のデータ数が必要となるが、そのようなクラスタ（通信状況の比較的良好であったクラスタ）が 1 つでも存在すれば、その復元データを利用することで、他のクラスタにおいては元データの個数分のデータを受信できれば元データを復元できることとなる。そして、部分的には、元データの個数よりもさらに少ない数の受信データからも元データを復元できることとなる。

## 【0036】

図9及び図10は、受信側端末20のデータ復元部22による元データの復元処理の動作を説明するフローチャートである。なお、図示の手順は、受信側端末20の動作の一例に過ぎず、必ずしもこの手順には限定されない。

図9、10に示すデータの復元処理に先立って、受信制御部21により送信側端末10から送信されたデータが受信されている。そして、データ復元部22はまず図9に示すように、エンコードされた順にクラスタをデコードして元データを復元する。そして、この手順で復元できなかったクラスタに対し、図10に示す手順で復元を試みる。

## 【0037】

図9を参照すると、データ復元部22は、受信制御部21にて受信された最初のクラスタであるクラスタ0を着目クラスタとし、その受信データに対して、予め設定された所定の初期値からなる $s$ 個の畳み込みデータ712を追加する（ステップ901）。この初期値は、送信側端末10の送信データ作成部12においてクラスタ0のパリティ・データを生成する際に元データ411に付加する畳み込みデータ412の初期値と同一である。この畳み込みデータ412、712の初期値は、予めネットワーク30に接続された全ての送信側端末10及び受信側端末20において共有することができるので、通信により喪失することなく追加できる。なお、最初にクラスタ0の受信データが元データを復元するための必要数（ $k + s$  個）だけあるかどうかを調べ、必要数に満たない場合にのみ畳み込みデータ712の初期値を追加するようにしても良い。

## 【0038】

次に、畳み込みデータ712の初期値を付加されたクラスタ0のデータ数が必要数の $k + s$  個以上あるかどうかを調べ、 $k + s$  個以上あれば、クラスタ0をデコードして元データ711を得る（ステップ902、903）。

畳み込みデータ712の初期値を付加されたクラスタ0のデータ数が $k + s$  個に満たない場合、データ復元部22は、次に直前のクラスタがデコード済みかどうかを調べる。クラスタ0の場合はその前に他のクラスタが存在しないので、デコード済みではないと判断し、当該クラスタ0を所定の記憶装置（例えば図2の



メインメモリ 103) に一時的に保持する (ステップ 904、908)。

ステップ 903 でクラスタ 0 をデコードできたか、もしくはステップ 908 でクラスタ 0 を保持した後、当該着目クラスタが最後のクラスタかどうかを調べ、次のクラスタがあれば、これを新たな着目クラスタとしてステップ 902 に戻る (ステップ 909、910)。

#### 【0039】

同様にして、新たな着目クラスタの受信データ数が  $k + s$  個以上あるかどうかを調べ、 $k + s$  個以上あれば、着目クラスタをデコードして元データ 711 及び畳み込みデータ 712 を得る (ステップ 902、903)。

着目クラスタの受信データ数が  $k + s$  個に満たない場合、データ復元部 22 は、次に直前のクラスタがデコード済みかどうかを調べる (ステップ 904)。そして、直前のクラスタがデコード済みであるならば、当該直前のクラスタにおける復元データをエンコードしたパリティ・データ 720 から  $s$  個までのデータ 722 を補充データとして取得し、補充する (ステップ 905)。その上で、データを補充された着目クラスタのデータ数が  $k + s$  個以上あるかどうかを改めて調べ、 $k + s$  個以上あれば、データが補完されたので着目クラスタをデコードして元データ 711 及び畳み込みデータ 712 を得る (ステップ 906、907)。

ステップ 904 で直前のクラスタがデコード済みでないと判断された場合、及び直前のクラスタから補充データを補充しても着目クラスタのデータ数が  $k + s$  個に満たなかった場合は、当該着目クラスタを所定の記憶装置 (例えば図 2 のメインメモリ 103) に一時的に保持する (ステップ 908)。

#### 【0040】

ステップ 903 またはステップ 907 で着目クラスタをデコードできたか、もしくはステップ 908 で着目クラスタを保持した後、当該着目クラスタが最後のクラスタかどうかを調べ、次のクラスタがあれば、これを新たな着目クラスタとしてステップ 902 に戻る (ステップ 909、910)。

着目クラスタが最後のクラスタであったならば、次にデータ復元部 22 は、ステップ 908 で保持した (デコードされていない) クラスタについて、図 10 による処理を行う。

## 【0041】

図10を参照すると、データ復元部22は、記憶装置に保持されているクラスタの1つに着目し（ステップ1001）、着目クラスタの直後のクラスタがデコード済みかどうかを調べる（ステップ1002）。直後のクラスタがデコード済みであるならば、そのデコードされた畳み込みデータ712を補充データとして取得し、補充する（ステップ1003）。そして、データを補充された着目クラスタのデータ数が $k + s$ 個以上あるかどうかを改めて調べ、 $k + s$ 個以上あれば、データが補完されたので着目クラスタをデコードして元データ711及び畳み込みデータ712を得る（ステップ1004、1005）。

以上の処理を、記憶装置に未処理のクラスタが無くなるまで繰り返す（ステップ1006）。また、直後のクラスタからデータを補充しても着目クラスタの $k + s$ 個に満たない場合は、着目クラスタに対するこれ以上のデータの補充はできないので、エラー処理（エラーメッセージの出力等）を行って処理を終了する（ステップ1007）。

## 【0042】

上記の動作において、ステップ1002で直後のクラスタがデコードされていないと判断した場合、このサイクルで当該着目クラスタはデコードされない。しかし、上記の処理を未処理のクラスタに対して逐次行っていく、当該クラスタの後方のクラスタがデコードされることで、当該クラスタが次に着目クラスタとなった際にデコードされることとなる。この他、図10に示した、直後のクラスタからデータを補充する操作を、クラスタを受信した順番の逆順に行っていくこととすれば、直後のクラスタが未だデコードされていないためにデータを補充できないということは無くなる（この場合、直後のクラスタがデコードされていないと判断されると、着目クラスタにデータの補充はできないので、エラー処理を行って処理を終了することとなる）。

## 【0043】

また、上記の手順は例示に過ぎず、他の手順によってデータを復元しても構わない。例えば $k + s$ 個のデータがあるクラスタについて先に元データ711及び畳み込みデータ712を復元してしまい、その後に $k + s$ 個の必要数に満たない

クラスタについて前後のクラスタからデータを補充して復元するといった手順とすることもできる。

あるいは、受信した各クラスタにおける元データやパリティの数を調べ、必要数以上有するクラスタと必要数に満たないクラスタとの位置関係及び補充すべきデータの数に基づいて、どのクラスタが前、あるいは後ろのクラスタにいくつデータを供給するか、及びデコード・エンコードの順番を決定する。そして、決定された順番にしたがってエンコード・デコードを行い、必要数に満たないクラスタに補充が必要な数だけのデータを得て、かかるクラスタの元データを復元するといった手順とすることもできる。

さらに、デコード・エンコードの順番を決定する際、エンコード・デコードの計算量やDisk I/Oの手間も考慮することができる。例えば、所定のクラスタ及びその前後のクラスタの受信データ数がいずれも必要数を満たしているならば、当該所定のクラスタにおいて畳み込みデータ712は復元する必要が無く、元データ711のみを復元する。前のクラスタで受信データが必要数よりも1個不足しているなら、当該所定のクラスタにおいて畳み込みデータ712を1個だけ復元する。所定のクラスタの受信データ数が必要数に不足し、その前後のクラスタの受信データが必要数を満たすならば、パリティ・データ720におけるデータ722を得るよりも畳み込みデータ712を得る方が処理に要する手間が少ないので、後のクラスタの畳み込みデータ712からデータを補充する。

#### 【0044】

以上のようにしてデータ復元部22により復元された元データ（及び受信制御部21にて受信された元データ）711は、送信側端末10において元のデータファイルをクラスタに分割したものである。したがって、かかる元データは、クラスタへの分割処理の逆変換により連結され、最終的に、送信側端末10においてファイル格納部11から読み出されたデータファイルに等しいデータファイルが復元されて、ファイル格納部23に格納される。

#### 【0045】

上記のように、本実施の形態は、データ送信の際の処理として、各クラスタに対し、元データに他のクラスタのデータ（パリティ・データ）を付加してエンコ

ードすることにより、パリティ・データを生成し、元データと共に送信する。したがって、受信データが喪失して元データを復元するための必要数に満たない場合でも、当該他のクラスタからデータを補充することによって、元データを復元することが可能となる。

この手法では、図 8 を参照して説明したように、通信におけるデータの喪失により、部分的に受信データの数元データの数よりもさらに少ない場合であっても元データの復元を可能とし、FEC (Forward Error Correction) によるデータ通信において、受信成功確率を高め、データ通信の信頼性の向上を図ることができる。

これは逆に、受信成功確率を一定とするならば、パリティ・データ（本実施の形態における送信用パリティ・データ）のデータ数を少なくできることを意味する。すなわち、所定の受信成功確率を満足するデータ通信における冗長度（パリティ・データの割合）を低く抑えることが可能となる。

#### 【0046】

上述した実施の形態では、送信データを作成する際、所定のクラスタにおけるパリティ・データ 420 のうち、送信用パリティ・データ 421 以外の  $s$  個のデータ 422 を次のクラスタの元データに付加している（図 4 参照）。したがって、受信側端末 20 では、最後のクラスタについては、その後方のクラスタからデータを補充することができない。すなわち、最後のクラスタのみ受信成功確率が低下することとなる。

そこで、最後のクラスタのみ、元データ 411 と送信用パリティ・データ 421 とからなる  $k+t$  個のデータを含む送信データに加えて、パリティ・データ 420 における送信用パリティ・データ 421 以外の  $s$  個のデータ 422 も送信する。このようにすれば、最後のクラスタについても他のクラスタと同様の補充用のデータが供給されることとなる。なお、最後のクラスタにおける  $s$  個のデータ 422 の送信は、当該クラスタのみ、元データ 411 にパリティ・データ 420 の全てを付加した  $k+t+s$  個のデータからなるデータ列を送信データとして送信するようにしても良いし、送信用パリティ・データ 421 を除く  $s$  個のデータ 422 を別途送信しても良い。さらに、必ずしも  $s$  個のデータを全て送信しなく

でも良く、 $t' > t$  なる  $t'$  個のデータを前記元データに付加して送信データを作成して送信することも可能である。

なお、各クラスタの元のデータファイルに対応する並び順や最初のクラスタ、最後のクラスタ等を識別するための情報は、各クラスタを送信する際のパケットのヘッダ等に記述しておき、受信側端末 20 においてこのパケットヘッダの記述に基づいて判断できるようにすることができる。

#### 【0047】

##### 【発明の効果】

以上説明したように、本発明によれば、FEC によるデータ通信において、配信成功確率を高め、データ通信の信頼性の向上を図ることができる。

また、本発明によれば、FEC によるデータ通信における冗長度を低く抑えることができる。

##### 【図面の簡単な説明】

【図1】 本実施の形態によるデータ通信が行われるネットワークシステムの概略構成を示す図である。

【図2】 図1に示した送信側端末及び受信側端末を実現するのに好適なコンピュータ装置のハードウェア構成の例を模式的に示した図である。

【図3】 本実施の形態における送信側端末の機能構成を示す図である。

【図4】 本実施の形態におけるパリティ・データの生成方法を説明する図である。

【図5】 本実施の形態の送信側端末によるデータ送信時の動作を説明するフローチャートである。

【図6】 本実施の形態における受信側端末の機能構成を示す図である。

【図7】 本実施の形態におけるデコード方法を説明する図である。

【図8】 本実施の形態におけるデコード済みのクラスタを利用したデータ補完の様子を説明する図である。

【図9】 本実施の形態の受信側端末によるデータ受信時の動作を説明するフローチャートである。

【図10】 本実施の形態の受信側端末によるデータ受信時の動作を説明す

るフローチャートである。

【図 11】 従来技術によるデータ通信方法を説明する図である。

【図 12】 従来技術によるパリティ・データの生成方法を示す図である。

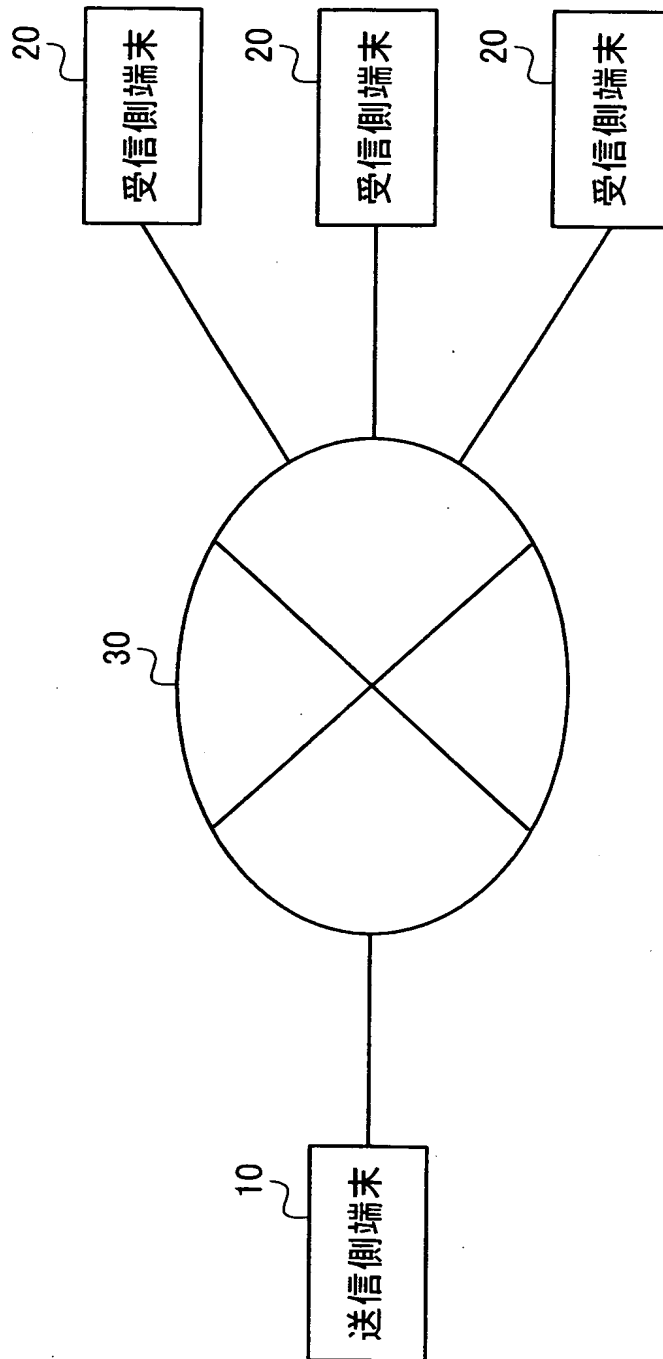
【符号の説明】

10…送信側端末、11…ファイル格納部、12…送信データ作成部、13…送信制御部、20…受信側端末、21…受信制御部、22…データ復元部、23…ファイル格納部、30…ネットワーク、101…CPU、102…M/Bチップセット、103…メインメモリ、105…ハードディスク、106…ネットワークインターフェイス

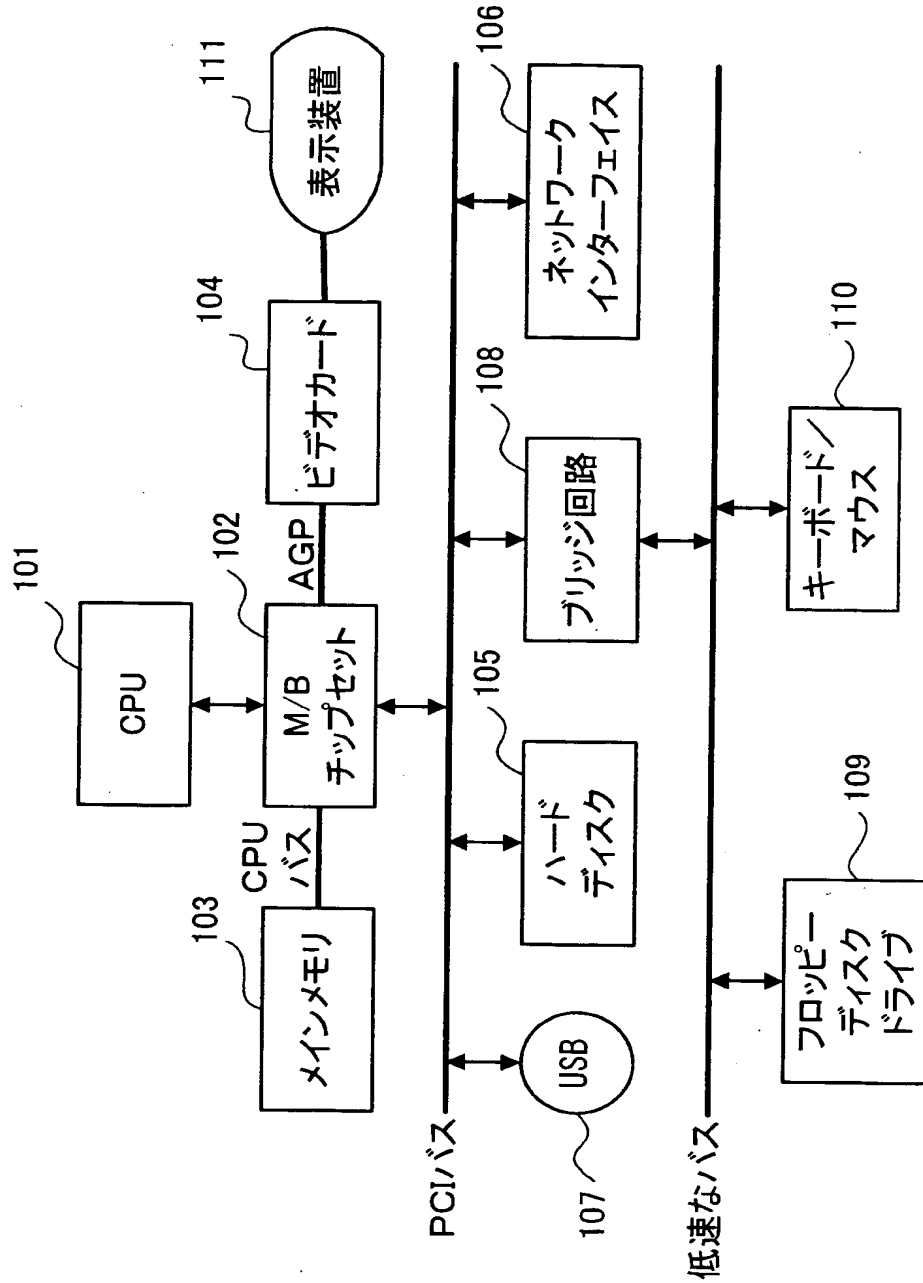
【書類名】

図面

【図 1】

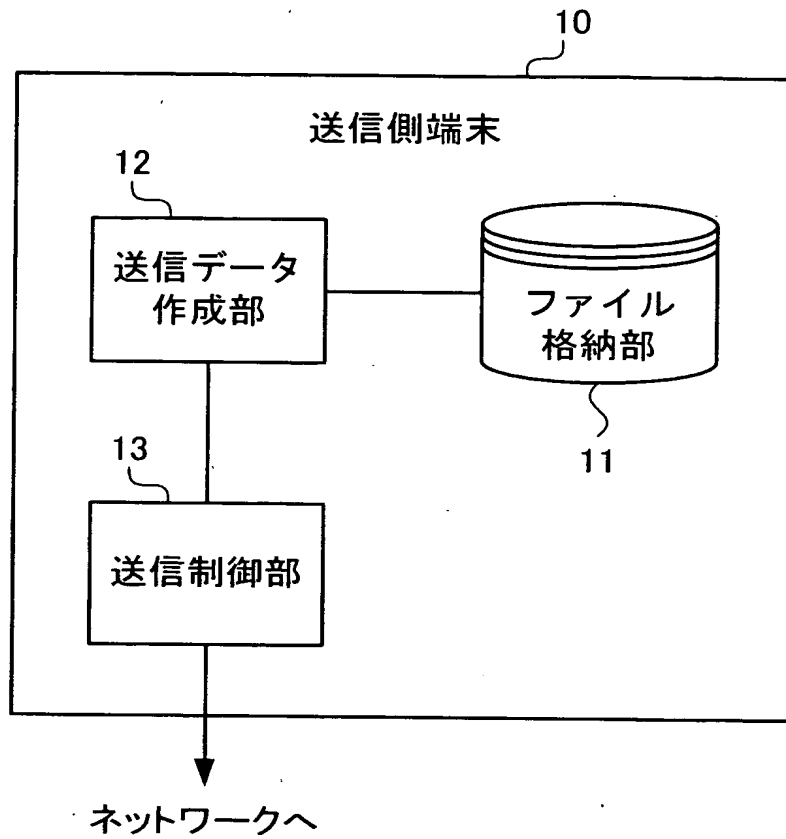


【図 2】

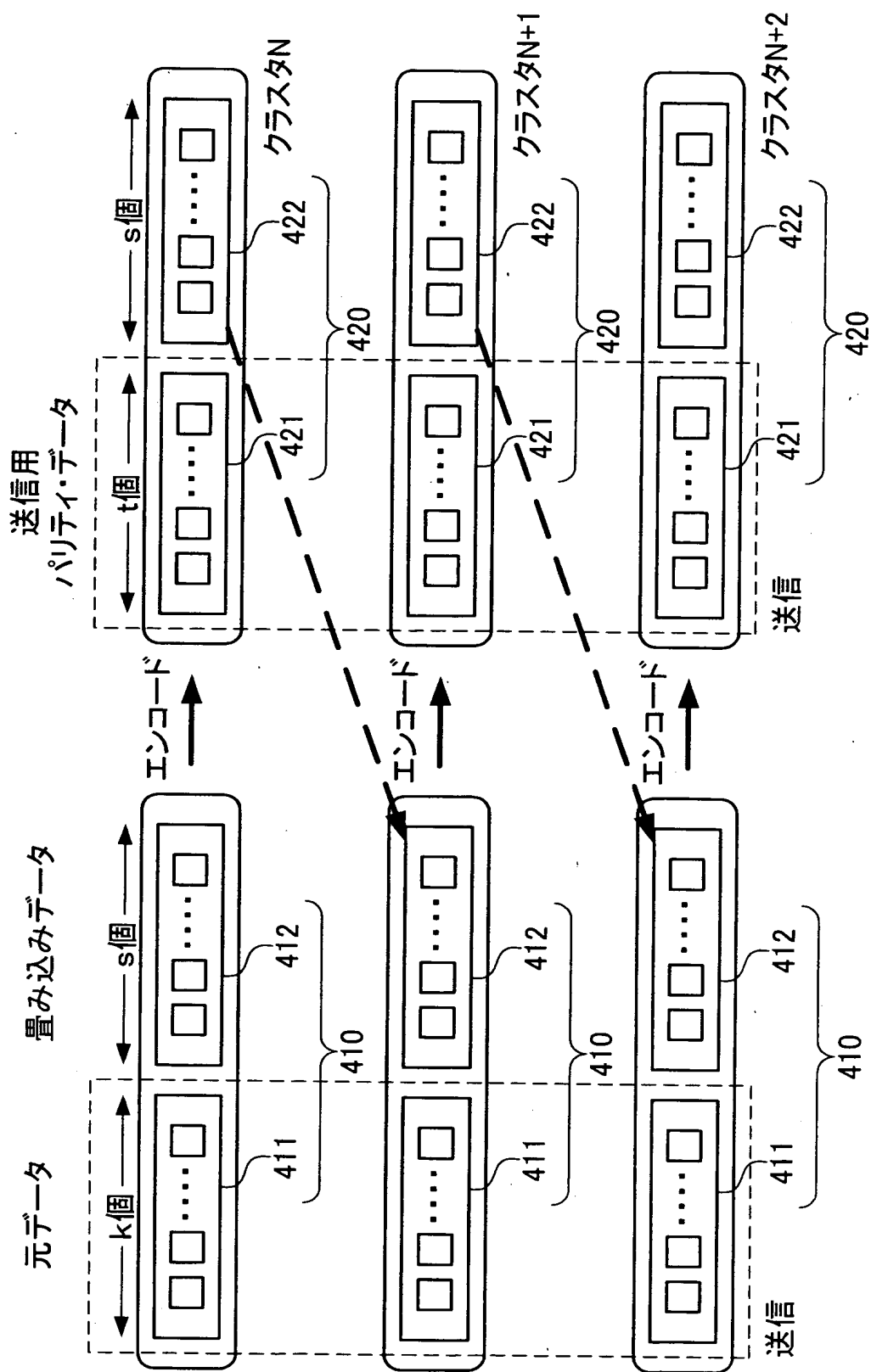




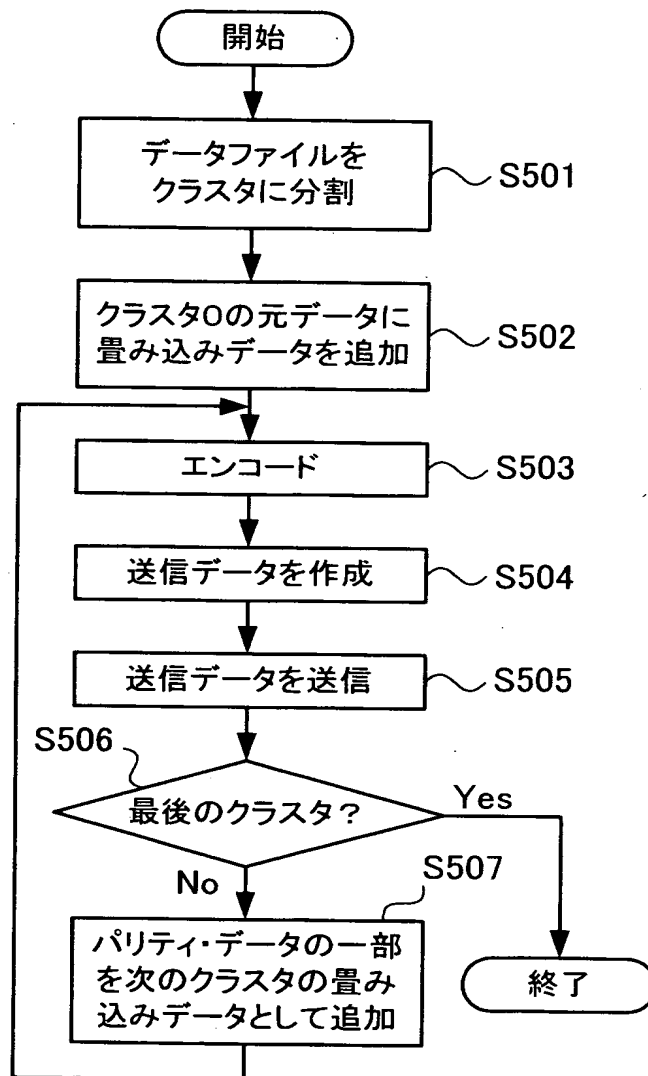
【図 3】



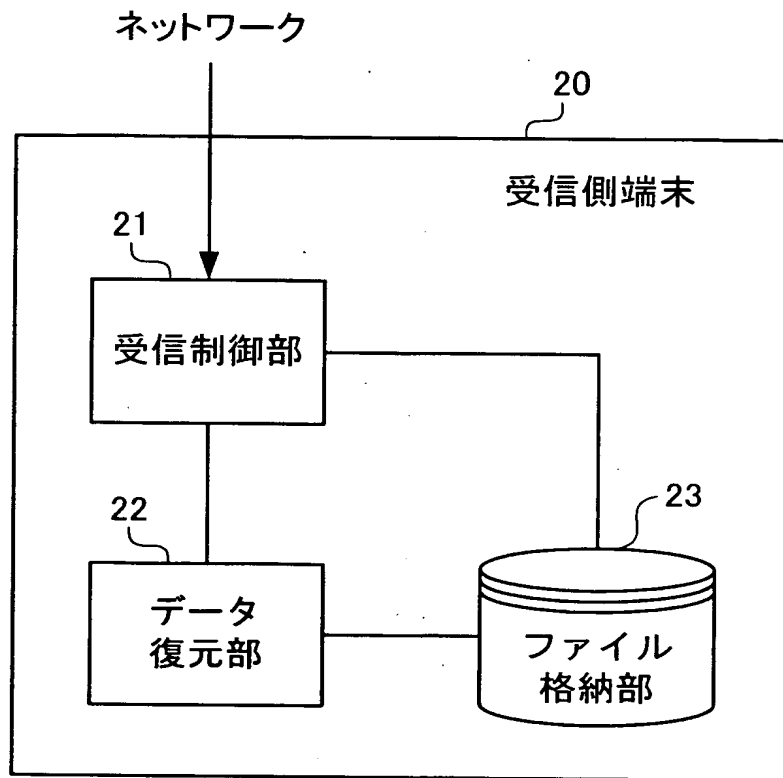
【図 4】



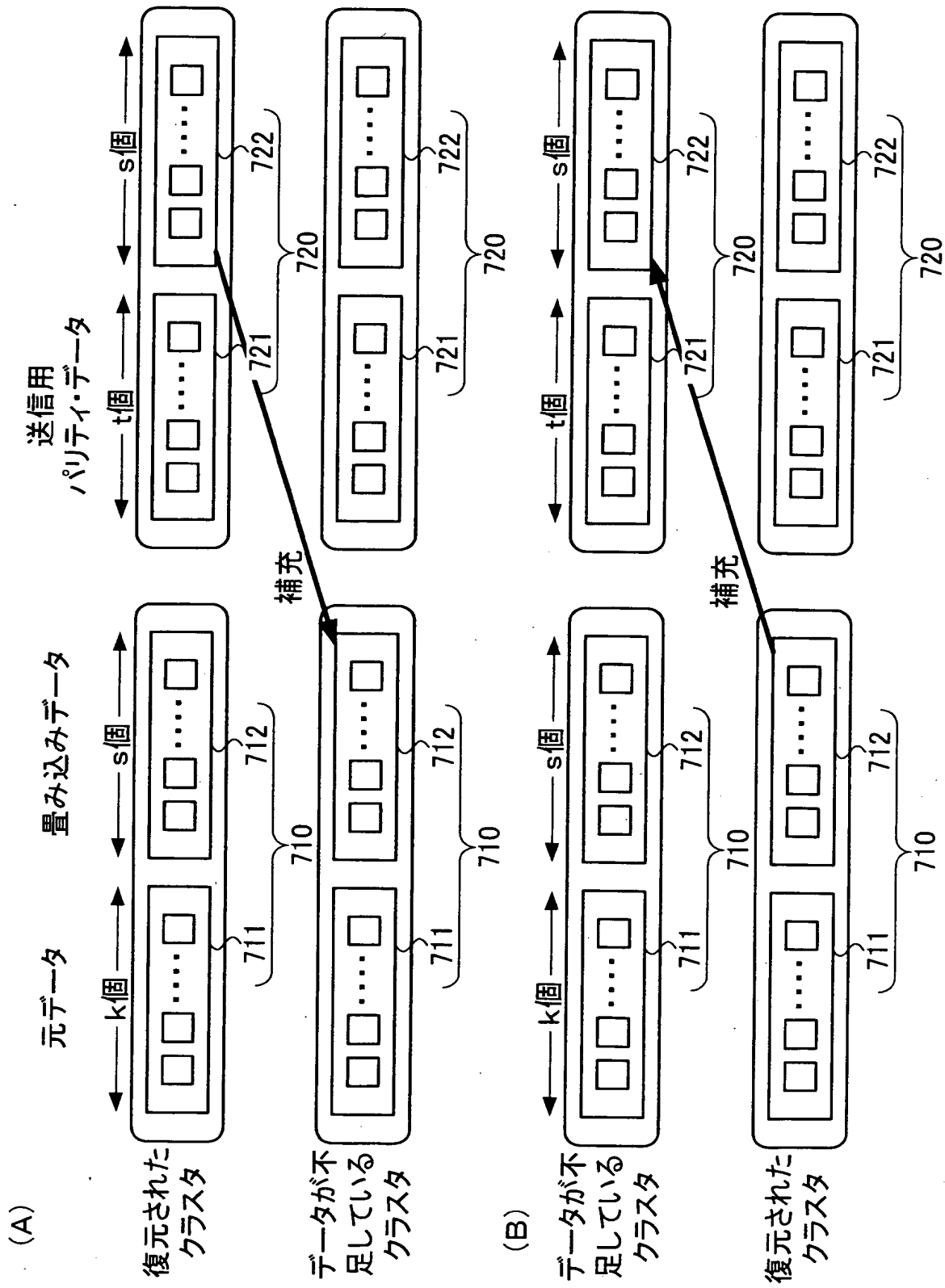
【図 5】



【図 6】

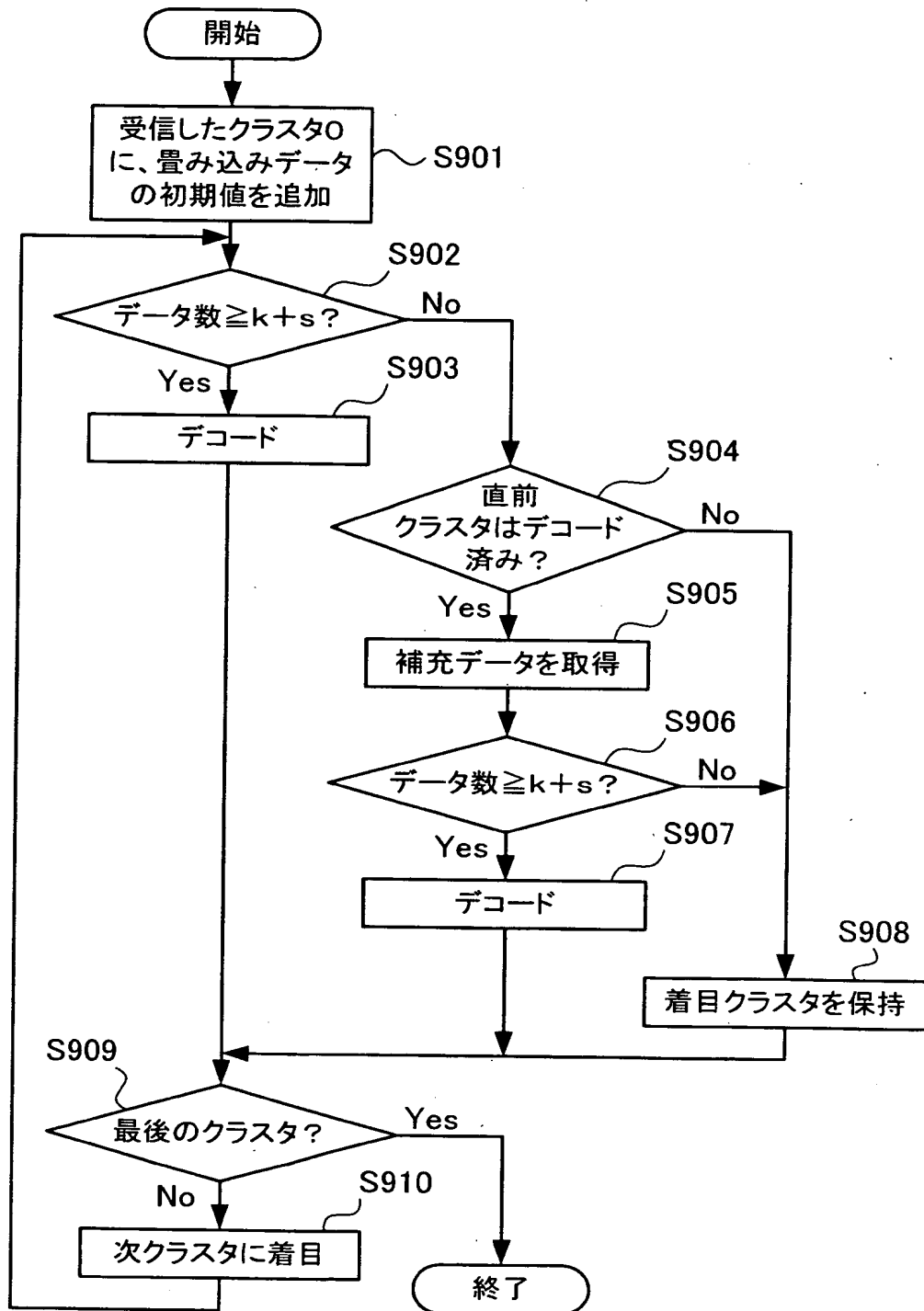


【図 7】

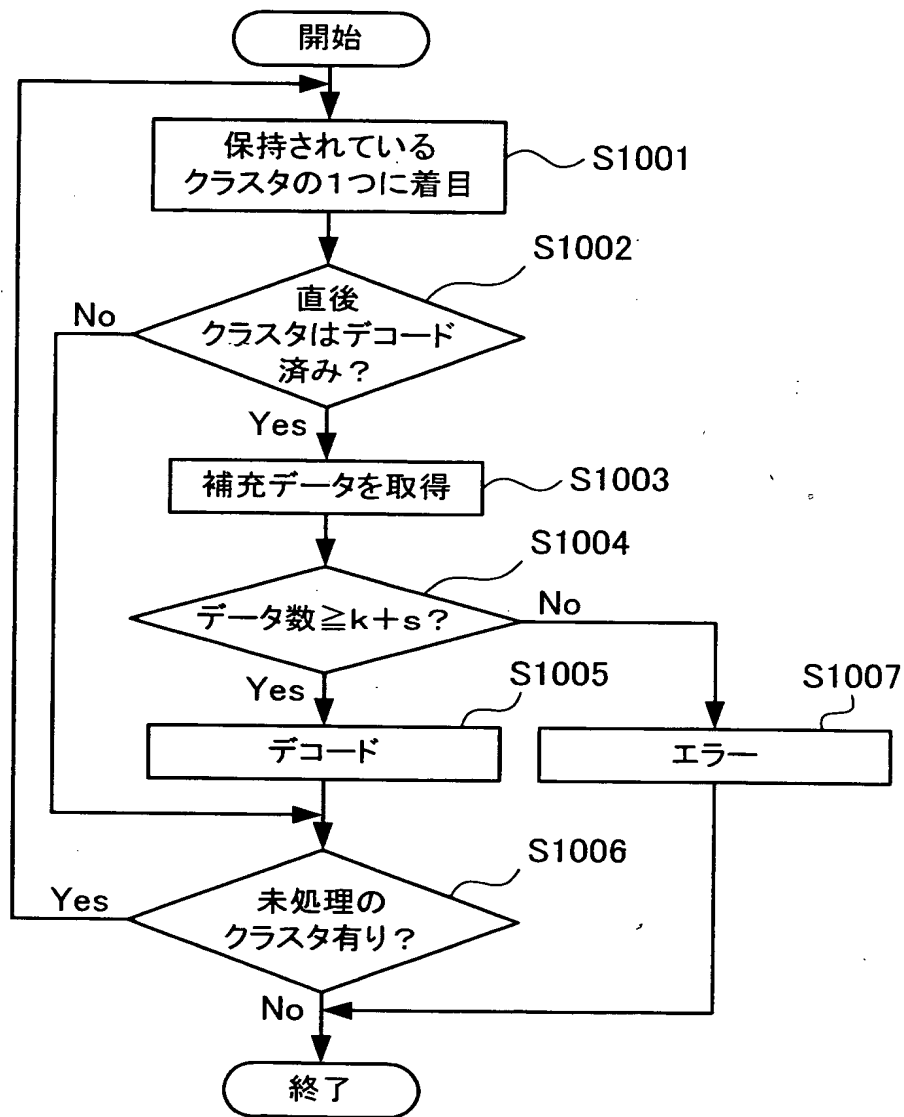




【図 9】

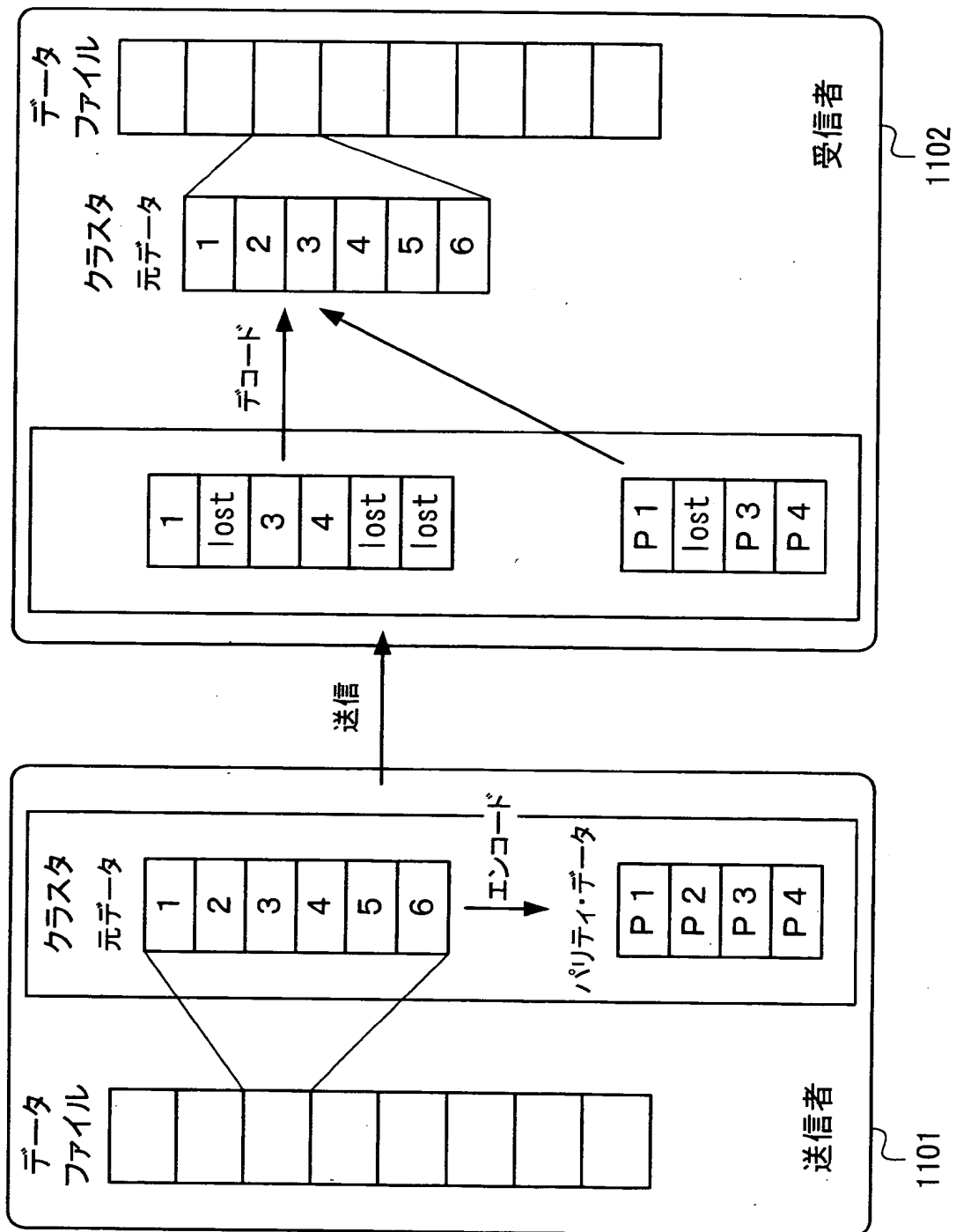


【図10】

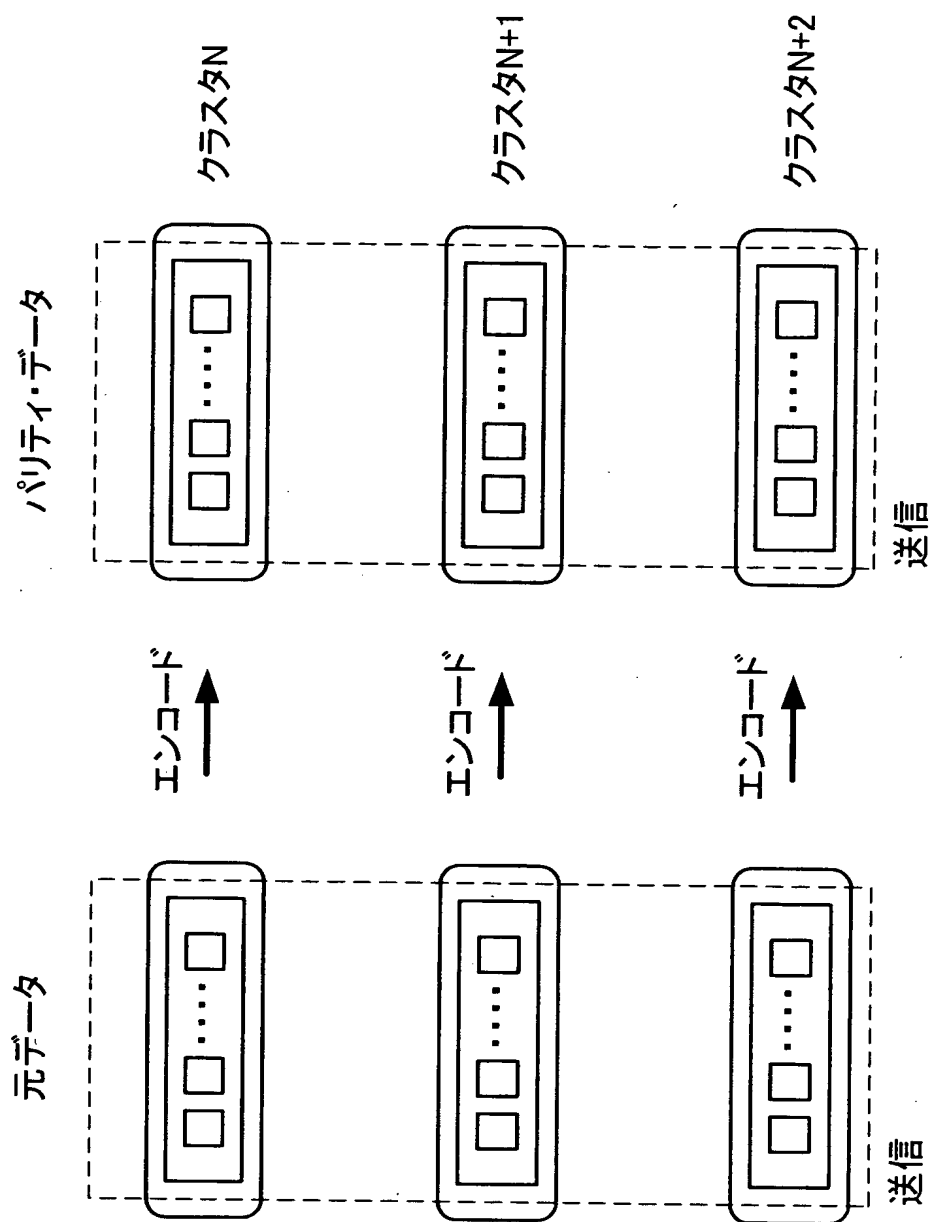




【図 11】



【図 12】



【書類名】 要約書

【要約】

【課題】 F E C によるデータ通信において、送信成功確立を高め、データ通信の信頼性の向上を図る。

【解決手段】 データを送信する際に送信しようとする元データのデータファイルを  $k$  個のデータからなるクラスタに分割する。そして、このクラスタの元データ 4 1 1 に他のクラスタから取得したデータによる畳み込みデータ 4 1 2 を加え、エンコードしてパリティ・データ 4 2 0 を生成する。このパリティ・データ 4 2 0 の一部を送信用パリティ・データ 4 2 1 として元データに付加し、送信データとしてネットワーク上の他の端末へ送信する。

【選択図】 図 4

## 認定・付加情報

特許出願の番号	特願 2002-345230
受付番号	50201800345
書類名	特許願
担当官	金井 邦仁 3072
作成日	平成 14 年 12 月 2 日

## &lt;認定情報・付加情報&gt;

## 【特許出願人】

【識別番号】	390009531
【住所又は居所】	アメリカ合衆国 10504、ニューヨーク州 アーモンク ニュー オーチャード ロード
【氏名又は名称】	インターナショナル・ビジネス・マシーンス・コーポレーション

## 【代理人】

【識別番号】	100086243
【住所又は居所】	神奈川県大和市下鶴間 1623 番地 14 日本アイ・ビー・エム株式会社 大和事業所内
【氏名又は名称】	坂口 博

## 【代理人】

【識別番号】	100091568
【住所又は居所】	神奈川県大和市下鶴間 1623 番地 14 日本アイ・ビー・エム株式会社 大和事業所内
【氏名又は名称】	市位 嘉宏

## 【代理人】

【識別番号】	100108501
【住所又は居所】	神奈川県大和市下鶴間 1623 番 14 日本アイ・ビー・エム株式会社 知的所有権
【氏名又は名称】	上野 剛史

## 【復代理人】

【識別番号】	100104880
【住所又は居所】	東京都港区赤坂 5-4-11 山口建設第 2 ビル 6 F セリオ国際特許事務所
【氏名又は名称】	古部 次郎

次頁無

特願 2002-345230

出 願 人 履 歴 情 報

識別番号

[390009531]

1. 変更年月日 2000年 5月16日  
[変更理由] 名称変更  
住 所 アメリカ合衆国10504、ニューヨーク州 アーモンク (番地なし)  
氏 名 インターナショナル・ビジネス・マシーンズ・コーポレーション
  
2. 変更年月日 2002年 6月 3日  
[変更理由] 住所変更  
住 所 アメリカ合衆国10504、ニューヨーク州 アーモンク ニュー オーチャード ロード  
氏 名 インターナショナル・ビジネス・マシーンズ・コーポレーション